METHOD AND APPARATUS FOR TRANSMITTING VIDEO AND GRAPHICS IN A COMPRESSED FORM

Cross-Reference to Related Application

This application is a continuation of co-pending United States Patent Application Serial No. 09/428,066, filed October 27, 1999, which application is incorporated by reference herein.

BACKGROUND OF THE DISCLOSURE

1. Field of the Invention

The invention relates to communications systems in general and, more specifically, the invention relates to a video compression technique suitable for use in an interactive multimedia information delivery system.

2. Description of the Background Art

Over the past few years, the television industry has seen a transformation in a variety of techniques by which its programming is distributed to consumers. Cable television systems are doubling or even tripling system bandwidth with the migration to hybrid fiber coax (HFC) cable transmission systems. Customers unwilling to subscribe to local cable systems have switched in high numbers to direct broadcast satellite (DBS) systems. And, a variety of other approaches have been attempted focusing primarily on high bandwidth digital technologies, intelligent two way set top boxes, or other methods of attempting to offer service differentiated from standard cable and over the air broadcast systems.

With this increase in bandwidth, the number of programming choices has also increased. Leveraging off the availability of more intelligent set top boxes, several companies have developed elaborate systems for providing an interactive listing of a vast array of channel offerings, expanded textual information about individual programs, the

ability to look forward to plan television viewing as much as several weeks in advance, and the option of automatically programming a video cassette recorder (VCR) to record a future broadcast of a television program.

Unfortunately, the existing program guides have several drawbacks. They tend to require a significant amount of memory, some of them needing upwards of one megabyte of memory at the set top terminal (STT). They are very slow to acquire their current database of programming information when they are turned on for the first time or are subsequently restarted (e.g., a large database may be downloaded to a STT using only a vertical blanking interval (VBI) data insertion technique). Disadvantageously, such slow database acquisition may result in out-of-date database information or, in the case of a pay-per-view (PPV) or video-on-demand (VOD) system, limited scheduling flexibility for the information provider.

Therefore, it is desirable to provide a data compression and decompression technique that enables interactive program guides having graphics and video portions to be efficiently transmitted through an interactive information distribution system.

SUMMARY OF THE INVENTION

The invention is an apparatus for compressing and transmitting both video and graphics portions of an interactive program guide (IPG). For an IPG that comprises a graphics portion and one or more video portions having audio associated with the video portions, the inventive system separately encodes the video portion and the graphics portion. The video portion is slice-base encoded using a predictive encoder, e.g., an MPEG encoder, that produces a bitstream comprising intra-coded picture slices and predictive-coded picture slices. The graphics portion is separately slice-base encoded to produce encoded slices

of the graphics image. The encoded slices of the graphics portion can be stored in a database and recalled as needed for transmission. To transmit an IPG, a transport stream is created containing the intra-coded and predictive-coded video streams as well as the encoded slices that comprise a graphics image that is to be included in the IPG. The receiver reassembles the components of the IPG by decoding the slice-based streams.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the present invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings, in which:

Figure 1 depicts an example of one frame of an interactive program guide (IPG) taken from a video sequence that can be encoded using the present invention;

Figure 2 depicts a block diagram of an illustrative interactive information distribution system that includes the encoding unit and process of the present invention;

Figure 3 depicts a slice map for the IPG of Figure 1;

Figure 4 depicts a block diagram of the encoding unit of Figure 2;

Figure 5 depicts a block diagram of the local neighborhood network of Figure 2;

Figure 6 depicts a matrix representation of program guide data with the data groupings shown for efficient encoding in accordance with the present invention;

Figure 7 is a diagrammatic flow diagram of a process for generating a portion of transport stream containing intra-coded video and graphics slices;

Figure 8 is a diagrammatic flow diagram of a process for generating a portion of transport stream containing predictive-coded video and graphics slices;

Figure 9 illustrates a data structure of a transport stream used to transmit the IPG of Figure 1;

Figure 10 is a diagrammatic flow diagram of an alternative process for generating a portion of transport stream containing predictive-coded video and graphics slices;

Figure 11A depicts an illustration of an IPG having a graphics portion and a plurality of video portions;

Figure 11B depicts a slice map for the IPG of Figure 11A;

Figure 12 is a diagrammatic flow diagram of a process for generating a portion of transport stream containing intra-coded video and graphics slices for an IPG having a graphics portion and a plurality of video portions;

Figure 13 is a diagrammatic flow diagram of a process for generating a portion of transport stream containing predictive-coded video and graphics slices for an IPG having a graphics portion and a plurality of video portions;

Figure 14 depicts a block diagram of a receiver within subscriber equipment suitable for use in an interactive information distribution system;

Figure 15 depicts a flow diagram of a first embodiment of a slice recombination process;

Figure 16 depicts a flow diagram of a second embodiment of a slice recombination process;

Figure 17 depicts a flow diagram of a third embodiment of a slice recombination process; and

Figure 18 depicts a flow diagram of a fourth embodiment of a slice recombination process.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common to the figures.

DETAILED DESCRIPTION

This invention is a system for generating, distributing and receiving a transport stream containing compressed video and graphics information. The invention is illustratively used to encode a plurality of interactive program guides (IPGs) that enable a user to interactively review, preview and select programming for a television system.

The invention uses compression techniques to reduce the amount of data to be transmitted and increase the speed of transmitting program guide information. As such, the data to be transmitted is compressed so that the available transmission bandwidth is used more efficiently. To transmit an IPG having both graphics and video, the invention separately encodes the graphics from the video such that the encoder associated with each portion of the IPG can be optimized to best encode the associated portion. The invention illustratively uses a slice-based, predictive encoding process that is based upon the Moving Pictures Experts Group (MPEG) standard known as MPEG-2. MPEG-2 is specified in the ISO/IEC standards 13818, which is incorporated herein by reference.

The above-referenced standard describes data processing and manipulation techniques that are well suited to the compression and delivery of video, audio and other information using fixed or variable rate digital communications systems. In particular, the above-referenced standard, and other "MPEG-like" standards and techniques, compress, illustratively, video information using intra-frame coding techniques (such as run-length coding, Huffman coding and the like) and inter-frame coding techniques (such as forward and backward predictive coding, motion compensation and the like). Specifically, in the

case of video processing systems, MPEG and MPEG-like video processing systems are characterized by prediction-based compression encoding of video frames with or without intra- and/or inter-frame motion compensation encoding.

To enhance error recovery, the MPEG-2 standard contemplates the use of a "slice layer" where a video frame is divided into one or more slices. A slice contains one or more contiguous sequence of macroblocks. The sequence begins and ends at any macroblock boundary within the frame. An MPEG-2 decoder, when provided a corrupted bitstream, uses the slice layer to avoid reproducing a completely corrupted frame. For example, if a corrupted bitstream is decoded and the decoder determines that the present slice is corrupted, the decoder skips to the next slice and begins decoding. As such, only a portion of the reproduced picture is corrupted.

The present invention uses the slice layer for the main purpose of flexible encoding and compression efficiency in a head end centric end-to-end system. A slice-based encoding system enables the graphics and video of an IPG to be efficiently coded and flexibly transmitted as described below. Consequently, a user can easily and rapidly move from one IPG page to another IPG page.

A. <u>An Exemplary Interactive Program Guide</u>

The present invention can be employed for compressing and transmitting various types of video frame sequences that contain graphics and video information, and is particularly useful in compressing and transmitting interactive program guides (IPG) where a portion of the IPG contains video (referred to herein as the video portion) and a portion of the IPG contains a programming guide grid (referred to herein as the guide portion or graphics portion). The present invention slice-based encodes the guide portion separately from the slice-based encoded video

portion, transmits the encoded portions within a transport stream, and reassembles the encoded portions to present a subscriber (or user) with a comprehensive IPG. Through the IPG, the subscriber can identify available programming and select various services provided by their information service provider.

Figure 1 depicts a frame from an illustrative IPG page 100. In this particular embodiment of an IPG, the guide grid information is contained in portion 102 (left half page) and the video information is contained in portion 101 (right half page). The IPG display 100 comprises a first 105A, second 105B and third 105C time slot objects, a plurality of channel content objects 110-1 through 110-8, a pair of channel indicator icons 141A, 141B, a video barker 120 (and associated audio barker), a cable system or provider logo 115, a program description region 150, a day of the week identification object 131, a time of day object 139, a next time slot icon 134, a temporal increment/decrement object 132, a "favorites" filter object 135, a "movies" filter object 136, a "kids" (i.e., juvenile) programming filter icon 137, a "sports" programming filter object 138 and a VOD programming icon 133. It should be noted that the day of the week object 131 and next time slot icon 134 may comprise independent objects (as depicted in Figure 1) or may be considered together as parts of a combined object.

A user may transition from one IPG page to another, where each page contains a different graphics portion 102, i.e., a different program guide graphics. The details regarding the encoding and decoding of a series of IPG pages in accordance with the present invention are provided below.

Details regarding the operation of the IPG page of Figure 1, the interaction of this page with other pages and with a user are described in commonly assigned US patent

application 09/359,560 filed July 22, 1999 which is hereby incorporated herein by reference.

B. System

Figure 2 depicts a high-level block diagram of an information distribution system 200, e.g., a video-on-demand system or digital cable system, that incorporates the present invention. The system 200 contains head end equipment (HEE) 202, local neighborhood equipment (LNE) 228, a distribution network 204 (e.g., hybrid fiber-coax network) and subscriber equipment (SE) 206. This form of information distribution system is disclosed in commonly assigned U.S. patent number 6,253,375, issued June 26, 2001. The system is known as DIVA™ provided by DIVA Systems Corporation.

The HEE 202 produces a plurality of digital streams that contain encoded information in illustratively MPEG-2 compressed format. These streams are modulated using a modulation technique that is compatible with a communications channel 230 that couples the HEE 202 to one or more LNE (in Figure 1, only one LNE 228 is depicted). The LNE 228 is illustratively geographically distant from the HEE 202. The LNE 228 selects data for subscribers in the LNE's neighborhood and remodulates the selected data in a format that is compatible with distribution network 204. Although the system 200 is depicted as having the HEE 202 and LNE 228 as separate components, those skilled in the art will realize that the functions of the LNE may be easily incorporated into the HEE202. It is also important to note that the presented slice-based encoding method is not constrained to physical location of any of the components. The subscriber equipment (SE) 206, at each subscriber location 206_1, 206_2, ..., 206_n, comprises a receiver 224 and a display 226. Upon receiving a stream, the subscriber equipment receiver 224 extracts the

information from the received signal and decodes the stream to produce the information on the display, i.e., produce a television program, IPG page, or other multimedia program.

In an interactive information distribution system such as the one described in commonly assigned U.S. patent number 6,253,375, the program streams are addressed to particular subscriber equipment locations that requested the information through an interactive menu. A related interactive menu structure for requesting video-on-demand is disclosed in commonly assigned U.S. patent number 6,208,335. Another example of interactive menu for requesting multimedia services is the interactive program guide (IPG) disclosed in commonly assigned U.S. patent application 60/093,891, filed in July 23, 1998.

To assist a subscriber (or other viewer) in selecting programming, the HEE 202 produces information that can be assembled to create an IPG such as that shown in FIG. 1. The HEE produces the components of the IPG as bitstreams that are compressed for transmission in accordance with the present invention.

A video source 214 supplies the video sequence for the video portion of the IPG to an encoding unit 216 of the present invention. Audio signals associated with the video sequence are supplied by an audio source 212 to the encoding and multiplexing unit 216. Additionally, a guide data source 232 provides program guide data to the encoding unit 216. This data is typically in a database format, where each entry describes a particular program by its title, presentation time, presentation date, descriptive information, channel, and program source.

The encoding unit 216 compresses a given video sequence into one or more elementary streams and the graphics produced from the guide data into one or more elementary streams. As described below with respect to Figure 4, the elementary streams are produced using a

slice-based encoding technique. The separate streams are coupled to the cable modem 222.

The streams are assembled into a transport stream that is then modulated by the cable modem 222 using a modulation format that is compatible with the head end communications channel 230. For example, the head end communications channel may be a fiber optic channel that carries high speed data from the HEE 202 to a plurality of LNE 228. The LNE 228 selects IPG page components that are applicable to its neighborhood and remodulates the selected data into a format that is compatible with a neighborhood distribution network 204. A detailed description of the LNE 228 is presented below with respect to Figure 5.

The subscriber equipment 206 contains a receiver 224 and a display 226 (e.g., a television). The receiver 224 demodulates the signals carried by the distribution network 204 and decodes the demodulated signals to extract the IPG pages from the stream. The details of the receiver 224 are described below with respect to Figure 14.

B. <u>Encoding Unit 216</u>

The system of the present invention is designed specifically to work in a slice-based ensemble encoding environment, where a plurality of bitstreams are generated to compress video information using a sliced-based technique. In the MPEG-2 standard, a "slice layer" may be created that divides a video frame into one or more "slices". Each slice includes one or more macroblocks, where the macroblocks are illustratively defined as rectangular groups of pixels that tile the entire frame, e.g., a frame may consist of 30 rows and 22 columns of macroblocks. Any slice may start at any macroblock location in a frame and extend from left to right and top to bottom through the frame. The stop point of a slice can be chosen to be any macroblock start or end boundary. The

slice layer syntax and its conventional use in forming an MPEG-2 bitstream is well known to those skilled in the art and shall not be described herein.

When the invention is used to encode an IPG comprising a graphics portion and a video portion, the slice-based technique separately encodes the video portion of the IPG and the grid graphics portion of the IPG. As such, the grid graphics portion and the video portion are represented by one or more different slices. FIG. 3 illustrates an exemplary slice division of an IPG 100 where the guide portion 102 and the video portion 101 are each divided into N slices (e.g., g/s1 through g/sN and v/s1 through v/sN). Each slice contains a plurality of macroblocks, e.g., 22 macroblocks total and 11 macroblocks in each portion. The slices in the graphics portion are pre-encoded to form a "slice form grid page" database that contains a plurality of encoded slices of the graphics portion. The encoding process can also be performed real-time during the broadcast process depending on the preferred system implementation. In this way, the graphics slices can be recalled from the database and flexibly combined with the separately encoded video slices to transmit the IPG to the LNE and, ultimately, to the subscribers. The LNE assembles the IPG data for the neighborhood as described below with respect to FIG. 5. Although the following description of the invention is presented within the context of an IPG, it is important to note that the method and apparatus of the invention is equally applicable to a broad range of applications, such as broadcast video on demand delivery,, e-commerce, internet video education services, and the like, where delivery of video sequences with common content is required.

As depicted in Figure 4, the encoding unit 216 receives a video sequence and an audio signal. The audio source comprises, illustratively, audio information that is

associated with a video portion in the video sequence such as an audio track associated with still or moving images. For example, in the case of a video sequence representing a movie trailer, the audio stream is derived from the source audio (e.g., music and voice-over) associated with the movie trailer.

The encoding unit 216 comprises video processor 400, a graphics processor 402 and a controller 404. The video processor 400 comprises a compositor unit 406 and an encoder unit 408. The compositor unit 406 combines a video sequence with advertising video, advertiser or service provider logos, still graphics, animation, or other video information. The encoder unit 408 comprises one or more video encoders 410, e.g., a real-time MPEG-2 encoder and an audio encoder 412, e.g., an AC-3 encoder. The encoder unit 408 produces one or more elementary streams containing slice-based encoded video and audio information.

The video sequence is coupled to a real time video encoder 410. The video encoder then forms a slice based bitstream, e.g., an MPEG-2 compliant bit stream, for the video portion of an IPG. For purposes of this discussion, it is assumed that the GOP structure consists of an I-picture followed by ten B-pictures, where a P-picture separates each group of two B-pictures (i.e., "I-B-B-P-B-B-P-B-B-P-B-B-P-B-B"), however, any GOP structure and size may be used in different configurations and applications.

The video encoder 410 "pads" the graphics portion (illustratively the left half portion of IPG) with null data. This null data is replaced by the graphics grid slices, at a later step, within LNE. Since the video encoder processes only motion video information, excluding the graphics data, it is optimized for motion video encoding.

The controller 404 manages the slice-based encoding process such that the video encoding process is time and

spatially synchronized with the grid encoding process. This is achieved by defining slice start and stop locations according to the objects in the IPG page layout and managing the encoding process as defined by the slices. That is, the controller selects the graphic slices to be included in the bitstream, as well as adjusts the slice boundaries.

The graphics portion of the IPG is separately encoded in the graphics processor 402. The processor 402 is supplied guide data from the guide data source (232 in Figure 2). Illustratively, the guide data is in a conventional database format containing program title, presentation date, presentation time, program descriptive information and the like. The guide data grid generator 414 formats the guide data into a "grid", e.g., having a vertical axis of program sources and a horizontal axis of time increments. One specific embodiment of the guide grid is depicted and discussed in detail above with respect to Figure 1.

The guide grid is a video frame that is encoded using a guide encoder 416 optimized for video with text and graphics content. The guide encoder 416, which can be implemented as software, slice-base encodes the guide data grid to produce one or more bitstreams that collectively represent the entire guide data grid. The guide encoder 416 is optimized to effectively encode the graphics and text content.

The controller 404 defines the start and stop macroblock locations for each slice. The result is a GOP structure having intra-coded pictures containing I-picture slices and predicted pictures containing B and P-picture slices. The I-pictures slices are separated from the predicted picture slices. Each encoded slice is separately stored in a slice form grid page database 418. The individual slices can be addressed and recalled from the

database 418 as required for transmission. The controller 404 controls the slice-based encoding process as well as manages the database 418.

D. Local Neighborhood Equipment (LNE) 228

FIG. 5 depicts a block diagram of the LNE 228. The LNE 228 comprises a cable modem 500, slice combiner 502, a multiplexer 504 and a digital video modulator 506. The LNE 228 is coupled illustratively via the cable modem to the HEE 202 and receives a transport stream containing the encoded video information and the encoded guide data grid information. The cable modem 500 demodulates the signal from the HEE 202 and extracts the MPEG slice information from the received signal. The slice combiner 502 combines the received video slices with the guide data slices in the order in which the decoder at receiver side can easily decode without further slice re-organization. The resultant combined slices are PID assigned and formed into an illustratively MPEG compliant transport stream(s) by multiplexer 504. The slice-combiner (scanner) and multiplexer operation is discussed in detail with respect to Figures 5-10. The transport stream is transmitted via a digital video modulator 506 to the distribution network 204.

The LNE 228 is programmed to extract particular information from the signal transmitted by the HEE 202., As such, the LNE can extract video and guide data grid slices that are targeted to the subscribers that are connected to the particular LNE. For example, the LNE 228 can extract specific channels for representation in the guide grid that are available to the subscribers connected to that particular LNE. As such, unavailable channels to a particular neighborhood would not be depicted in a subscriber's IPG. Additionally, the IPG can contain targeted advertising, e-commerce, program notes, and the

like. As such, each LNE can combine different guide data slices with different video to produce IPG screens that are prepared specifically for the subscribers connected to that particular LNE. Other LNEs would select different IPG component information that is relevant to their associated subscribers.

Figure 6 illustrates a matrix representation 600 of a series of IPG pages. In the illustrated example, ten different IPG pages are available at any one time period, e.g., t_1, t_2, and so on. Each page is represented by a guide portion (g) and a common video portion (v) such that a first IPG page is represented by g1/v1, the second IPG page is represented by g2/v1 and so on. In the illustrative matrix 600, ten identical guide portions (g1-g10) are associated with a first video portion (v1). Each portion is slice-base encoded as described above within the encoding unit (216 of FIG.4).

Figure 6 illustrates the assignment of PIDs to the various portions of the IPG pages. In the figure, only the content that is assigned a PID is delivered to a receiver. The intra-coded guide portion slices g1 through g10 are assigned to PID1 through PID10 respectively. One of the common intra-coded video portion v1, illustratively the tenth IPG page, is assigned to PID11. In this form, substantial bandwidth saving is achieved by delivering intra-coded video portion slices v1 only one time. Lastly, the predictive-coded slices g1/v2 through g1/v15 are assigned to PID11. As shown in the figure, a substantial bandwidth saving is achieved by transmitting only one group of illustratively fourteen predicted picture slices, g1/v2 to g1/v15. This is provided by the fact that the prediction error images for each IPG page 1 to 10 through time units t2 to t15 contain the same residual images. Further details of PID assignment process is discussed in next sections.

Figure 7 depicts a process 700 that is used to form a bitstream 710 containing all the intra-coded slices encoded at a particular time t1 of Figure 6. At step 702, a plurality of IPG pages 702_1 through 702_{10} are provided to the encoding unit. At step 704, each page is slice base encoded to form, for example, guide portion slices g1/s1 through g1/sN and video portion slices v/s1 through v/sN for IPG page 1 704_1. The slice based encoding process for video and guide portions can be performed in different forms. For example, guide portion slices can be pre-encoded by a software MPEG-2 encoder or encoded by the same encoder as utilized for encoding the video portion. If the same encoder is employed, the parameters of the encoding process is adjusted dynamically for both portions. It is important to note that regardless of the encoder selection and parameter adjustment, each portion is encoded independently. While encoding the video portion, the encoding is performed by assuming the full frame size (covering both guide and video portions) and the guide portion of the full frame is padded with null data. This step, step 704, is performed at the HEE. At step 706, the encoded video and guide portion slices are sent to the LNE. If the LNE functionality is implemented as part of the HEE, then, the slices are delivered to the LNE as packetized elementary stream format or any similar format as output of the video encoders. If LNE is implemented as a remote network equipment, the encoded slices are formatted in a form to be delivered over a network via a preferred method such as cable modem protocol or any other preferred method. Once the slice-based streams are available in the LNE, the slice combiner at step 706 orders the slices in a form suitable for the decoding method at the receiver equipment. As depicted in Figure 7 (b), the guide portion and video portion slices are ordered in a manner as if the original pictures in Figure 7 (a) are scanned from left to

right and top to bottom order. Each of the slice packets are then assigned PID's as discussed in Figure 6 by the multiplexer; PID1 is assigned to g1/s1 … g1/sn, PID2 to g2/s1 … g2/sn, ..., PID10 to g10/s1 … g10/sn, and PID11 is assigned to v/s1 … v/sn. The resultant transport stream containing the intra-coded slices of video and guide portions is illustrated in Figure 7 (c). Note that based on this transport stream structure, a receiving terminal as discussed in later parts of this description of the invention, retrieves the original picture by constructing the video frames row-by-row, first retrieving, assuming PID1 is desired, e.g., g1/s1 of PID1 then v/s1 of PID11, next g1/s2 of PID1 then v/s2 of PID11 and so on.

Figure 8 illustrates a process 800 for producing a bitstream 808 containing the slices from the predictive-coded pictures accompanying the transport stream generation process discussed in Figure 7 for intra-coded slices. As shown in Figure 6, illustratively, only the predicted slices belonging to IPG page 1 is delivered. Following the same arguments of encoding process in Figure 7, at step 802, the predictive-coded slices are generated at the HEE independently and then forwarded to an LNE either as local or in a remote network location. At step 804, slices in the predictive-coded guide and video portion slices, illustratively from time periods t2 to t15, are scanned from left to right and top to bottom in slice-combiner and complete data is assigned PID 11 by the multiplexer. Note that the guide portion slices g1/s1 to g1/sn at each time period t2 to t15 does not change from their intra-coded corresponding values at t1. Therefore, these slices are coded as skipped macroblocks "sK". Conventional encoder systems do not necessarily skip macroblocks in a region even when there is no change from picture to picture. In order to provide this functionality, the encoder is given the parameters for discussed slices to skip macroblocks

without any further encoding evaluations. At step 806, the slice packets are ordered into a portion of final transport stream, first including the video slice packets v2/s1 … v2/SN to v15/s1 … v15/sN, then including the skipped guide slices sK/s1 … sK/sN from t2 to t15 in the final transport stream. FIG. 9 depicts a complete MPEG compliant transport stream 900 that contains the complete information needed by a decoder to recreate IPG pages that are encoded in accordance with the invention. The transport stream 900 comprises the intra-coded bitstream 710 of the guide and video slices (PIDS1 to 11), a plurality of audio packets 902 identified by an audio PID, and the bitstream 806 containing the predictive-coded slices in PID11. The rate of audio packet insertion between video packets is decided based on the audio and video sampling ratios. For example, if audio is digitally sampled as one tenth of video signal, then an audio packet may be introduced into the transport stream every ten video packets. The transport stream 900 may also contain, illustratively after every 64 packets, data packets that carry to the set top terminal overlay updates, raw data, HTML, java, URL, instructions to load other applications, user interaction routines, and the like. The data PIDs are assigned to different set of data packets related to guide portion slice sets and also video portion slice sets.

FIG. 10 illustrates a process 1000, an alternative embodiment of process 800 depicted in Figure 8, for producing a predictive-coded slice bitstream 1006. The process 1000, at step 1002, produces the slice base encoded predictive-coded slices. At step 1004, the slices are scanned to intersperse the "skipped" slices (s_K) with the video slices (v_1). The previous embodiment scanned the skipped guide portion and video portion separately. In this embodiment, each slice is scanned left to right and top to bottom completely, including the skipped guide and

video data. As such, at step 1008, the bitstream 1006 has the skipped guide and video slices distributed uniformly throughout the transport stream.

The foregoing embodiments of the invention assumed that the IPG page was divided into one guide portion and one video portion. For example, in Figure 1, the guide portion is the left half of the IPG page and the video portion is the right half of the IPG page. However, the invention can be extended to have a guide portion and multiple video portions, e.g., three. Each of the video portions may contain video having different rates of motion, e.g., portion one may run at 30 frames per second, portions two and three may run at 2 frames per second. Figure 11A illustrates an exemplary embodiment of an IPG 1100 having a guide portion 1102 and three video portions 1104, 1106 and 1108. To encode such an IPG, each portion is separately encoded and assigned PIDs. Figure 11B illustrates an assignment map for encoding each portion of the IPG page of Figure 11A. The guide portion 1002 is encoded as slices g/s1 through g/sN, while the first video portion 1004 is encoded as slices v/s1 through v/sM, and the second video portion 1006 is encoded as slices j/sM+1 through j/sL, the third video portion 1008 is encoded as slices p/sL+1 through p/sN.

Figure 12 depicts the scanning process 1200 used to produce a bitstream 1210 containing the intra-coded slices. The scanning process 1200 flows from left to right, top to bottom through the assigned slices of Figure 11B. PIDs are assigned, at step 1202, to slices 1 to M; at step 1204, to slices M+1 to L; and, at step 1206, to slices L+1 to N. As the encoded IPG is scanned, the PIDS are assigned to each of the slices. The guide portion slices are assigned PIDS 1 through 10, while the first video portion slices are assigned PID11, the second video portion slices are assigned PID12 and the third video portion slices are

assigned PID13. The resulting video portion of the bitstream 1210 contains the PIDS for slices 1-M, followed by PIDS for slices M+1 to L, and lastly by the PIDS for L+1 to N.

Figure 13 depicts a diagrammatical illustration of a process 1300 for assigning PIDS to the predictive-coded slices for the IPG of FIG. 11A. The scanning process 1300 is performed, at step 1302, from left to right, top to bottom through the V, J and P predicted encoded slices and PIDS are assigned where the V slices are assigned PID11, the J slices are assigned PID 12 and the P slices are assigned PID13. After the video portion predicted encoded slices have assigned PIDs, the process 1300, at step 1304, assigns PIDs to the skipped slices. The skipped guide slices vertically corresponding to the V slices are assigned PID11, the skipped slices vertically corresponding to the J slices are assigned PID12 and the skipped slices vertically corresponding to the P slices are assigned PID13. At step 1308, the resulting predictive-coded bitstream 1312 comprises the predicted video slices in portion 1306 and the skipped slices 1310. The bitstream 1210 of intra-coded slices and the bitstream 1312 of predictive-coded slices are combined into a transport stream having a form similar to that depicted in Figure 9.

Guide pages (video PIDs for groups of slices) can not be changed seamlessly using a standard channel change by the receiver switching from PID to PID directly, because such an operation flushes the video and audio buffers and typically gives half a second blank screen.

To have seamless decoder switching, a splice countdown (or random access indicator) packet is employed at the end of each video sequence to indicate the point at which the video should be switched from one PID to another.

Using the same profile and constant bit rate coding for the graphics encoding units, the generated streams for

different IPG pages are formed in a similar length compared to each other. This is due to the fact that guide pages are almost identical differing only in the characters. . In this way, while streams are generated having nearly identical lengths, the streams are not exactly the same length. Thus, a finer adjustment is required to synchronize the beginnings and ends of each sequence across all guide slices in order for the countdown switching to work.

The invention provides the act of synchronization of a plurality of streams that provides seamless switching at the receiver.

Three methods are provided for that purpose:

First, for each sequence the multiplexer in the LNE identifies the length of the longest guide page for that particular sequence, and then adds sufficient null packets to the end of each other guide page so that all the encoded guide page slices become the same length. Then, the multiplexer adds the switching packets at the end of the sequence, after all the null packets.

The second method requires buffering of all the packets for all guide page slices for each sequence. If this is allowed in the considered system, then the packets can be ordered in the transport stream such that the packets for each guide page slice appear at slightly higher or lower frequencies, so that they all finish at the same point. Then, the switching packets are added by the multiplexer in the LNE at the end of each stream without the null padding.

A third method is to start each sequence together, and then wait until all the packets for all the guide page slices have been generated. Once the generation of all packets is completed, switching packets are placed in the streams at the same time and point in each stream.

Depending on the implementation of decoder units within the receiver and requirements of the considered

application, each one of the methods can be applied with advantages. For example, the first method, which is null-padding, can be applied to avoid bursts of N packets of the same PID into a decoder's video buffer faster than the MPEG specified rate (e.g., 1.5 Mbit).

The teachings of the above three methods can be extended apply to similar synchronization problems and to derive similar methods for ensuring synchronization during stream switching.

E. Receiver 224

Figure 14 depicts a block diagram of the receiver 224 (also known as a set top terminal (STT) or user terminal) suitable for use in producing a display of an IPG in accordance with the present invention. The STT 224 comprises a tuner 1410, a demodulator 1420, a transport demultiplexer 1430, an audio decoder 1440, a video decoder 1450, an on-screen display processor (OSD) 1460, a frame store memory 1462, a video compositor 1490 and a controller 1470. User interaction is provided via a remote control unit 1480. Tuner 1410 receives, e.g., a radio frequency (RF) signal comprising, for example, a plurality of quadrature amplitude modulated (QAM) information signals from a downstream (forward) channel. Tuner 1410, in response to a control signal TUNE, tunes a particular one of the QAM information signals to produce an intermediate frequency (IF) information signal. Demodulator 1420 receives and demodulates the intermediate frequency QAM information signal to produce an information stream, illustratively an MPEG transport stream. The MPEG transport stream is coupled to a transport stream demultiplexer 1430.

Transport stream demultiplexer 1430, in response to a control signal TD produced by controller 1470, demultiplexes (i.e., extracts) an audio information stream

A and a video information stream V. The audio information stream A is coupled to audio decoder 1440, which decodes the audio information stream and presents the decoded audio information stream to an audio processor (not shown) for subsequent presentation. The video stream V is coupled to the video decoder 1450, which decodes the compressed video stream V to produce an uncompressed video stream VD that is coupled to the video compositor 1490. OSD 1460, in response to a control signal OSD produced by controller 1470, produces a graphical overlay signal VOSD that is coupled to the video compositor 1490. During transitions between streams representing the user interfaces, buffers in the decoder are not reset. As such, the user interfaces seamlessly transition from one screen to another.

The video compositor 1490 merges the graphical overlay signal VOSD and the uncompressed video stream VD to produce a modified video stream (i.e., the underlying video images with the graphical overlay) that is coupled to the frame store unit 1462. The frame store unit 1462 stores the modified video stream on a frame-by-frame basis according to the frame rate of the video stream. Frame store unit 1462 provides the stored video frames to a video processor (not shown) for subsequent processing and presentation on a display device.

Controller 1470 comprises a microprocessor 1472, an input/output module 1474, a memory 1476, an infrared (IR) receiver 1475 and support circuitry 1478. The microprocessor 1472 cooperates with conventional support circuitry 1478 such as power supplies, clock circuits, cache memory and the like as well as circuits that assist in executing the software routines that are stored in memory 1476. The controller 1470 also contains input/output circuitry 1474 that forms an interface between the controller 1470 and the tuner 1410, the transport demultiplexer 1430, the onscreen display unit 1460, the

back channel modulator 1495, and the remote control unit 1480. Although the controller 1470 is depicted as a general purpose computer that is programmed to perform specific interactive program guide control function in accordance with the present invention, the invention can be implemented in hardware as an application specific integrated circuit (ASIC). As such, the process steps described herein are intended to be broadly interpreted as being equivalently performed by software, hardware, or a combination thereof.

In the exemplary embodiment of Figure 14, the remote control unit 1480 comprises an 8-position joy stick, a numeric pad, a "select" key, a "freeze" key and a "return" key. User manipulations of the joy stick or keys of the remote control device are transmitted to a controller via an infra red (IR) link. The controller 1470 is responsive to such user manipulations and executes related user interaction routines 1400, uses particular overlays that are available in an overlay storage 1479.

After the signal is tuned and demodulated, the video streams are recombined via stream processing routine 1402 to form the video sequences that were originally compressed. The processing unit 1402 employs a variety of methods to recombine the slice-based streams, including, using PID filter 1404, demultiplexer 1430, as discussed in the next sections of this disclosure of the invention. Note that the PID filter implemented illustratively as part of the demodulator is utilized to filter the undesired PIDs and retrieve the desired PIDs from the transport stream. The packets to be extracted and decoded to form a particular IPG are identified by a PID mapping table (PMT) 1477. After the stream processing unit 1402 has processed the streams into the correct order (assuming the correct order was not produced in the LNE), the slices are sent to the MPEG decoder 1450 to generate the original uncompressed

IPG pages. If an exemplary transport stream with two PIDs as discussed in previous parts of the this disclosure, excluding data and audio streams, is received, then the purpose of the stream processing unit 1402 is to recombine the intra-coded slices with their corresponding predictive-coded slices in the correct order before the recombined streams are coupled to the video decoder. This complete process is implemented as software or hardware. In the illustrated IPG page slice structure, only one slice is assigned per row and each row is divided into two portions, therefore, each slice is divided into guide portion and video portion. In order for the receiving terminal to reconstruct the original video frames, one method is to construct a first row from its two slices in the correct order by retrieving two corresponding slices from the transport stream, then construct a second row from its two slices, and so on. For this purpose, a receiver is required to process two PIDs in a time period. The PID filter can be programmed to pass two desired PIDs and filter out the undesired PIDs. The desired PIDs are identified by the controller 1472 after the user selects an IPG page to review. A PID mapping table (1477 of Figure 14) is accessed by the controller 1472 to identify which PIDS are associated with the desired IPG. If a PID filter is available in the receiver terminal, then it is utilized to receive two PIDs containing slices for guide and video portions. The demultiplexer then extracts packets from these two PIDs and couples the packets to the video decoder in the order in which they arrived. If the receiver does not have an optional PID filter, then the demultiplexer performs the two PID filtering and extracting functions. Depending on the preferred receiver implementation, the following methods are provided in Figures 15-18 to recombine and decode slice-based streams.

E1. Recombination Method 1

In this first method, intra-coded slice-based streams (I-streams) and the predictive-coded slice-based streams (PRED streams) to be recombined keep their separate PID's until the point where they must be depacketized. The recombination process is conducted within the demultiplexer 1430 of the subscriber equipment For illustrative purposes, assuming a multi-program transport stream with each program consisting of I-PIDs for each intra-coded guide slice, I-PIDs for the intra-coded video slices, one PRED-PID for predicted guide and video, an audio-PID, and multiple data-PIDs, any packet with a PID that matches any of the PID's within the desired program (as identified in a program mapping table) are depacketized and the payload is sent to the elementary stream video decoder. Payloads are sent to the decoder in exactly in the order in which the packets arrive at the demultiplexer.

Figure 15 is a flow diagram of the first packet extraction method 1500. The method starts at step 1505 and proceeds to step 1510 to wait for (user) selection of an I-PID to be received. The I-PID, as the first picture of a stream's GOP, represents the stream to be received. However, since the slice-based encoding technique assigns two or more I-PIDS to the stream (i.e., I-PIDs for the guide portion and for one or more video portions), the method must identify two or more I-PIDs. Upon detecting a transport packet having the selected I-PIDs, the method 1500 proceeds to step 1515.

At step 1515, the I-PID packets (e.g., packets having PID-1 and PID-11) are extracted from the transport stream, including the header information and data, until the next picture start code. The header information within the first-received I-PID access unit includes sequence header, sequence extension, group start code, GOP header, picture header, and picture extension, which are known to a reader

that is skilled in MPEG-1 and MPEG-2 compression standards. The header information in the next I-PID access units that belongs to the second and later GOP's includes group start code, picture start code, picture header, and extension. The method 1500 then proceeds to step 1520 where the payloads of the packets that includes header information related to video stream and I-picture data are coupled to the video decoder 1550 as video information stream V. The method 1500 then proceeds to step 1525.

At step 1525, the predicted picture slice-based stream packets PRED-PID, illustratively the PID-11 packets of fourteen predicted pictures in a GOP of size fifteen, are extracted from the transport stream. At step 1530, the payloads of the packets that includes header information related to video stream and predicted-picture data are coupled to the video decoder 1550 as video information stream V. At the end of step 1530, a complete GOP, including the I-picture and the predicted-picture slices, are available to the video decoder 1550. As the payloads are sent to the decoder in exactly in the order in which the packets arrive at the demultiplexer, the video decoder decodes the recombined stream with no additional recombination process. The method 1500 then proceeds to step 1535.

At step 1535, a query is made as to whether a different I-PID is requested, e.g., new IPG is selected. If the query at step 1535 is answered negatively, then the method 1500 proceeds to step 1510 where the transport demultiplexer 1530 waits for the next packets having the PID of the desired I-picture slices. If the query at step 1535 is answered affirmatively, then the PID of the new desired I-picture slices is identified at step 1540 and the method 1500 returns to step 1510.

The method 1500 of Figure 15 is used to produce a conformant MPEG video stream V by concatenating a desired

I-picture slices and a plurality of P- and/or B-picture slices forming a pre-defined GOP structure.

E2. Recombination Method 2

The second method of recombining the video stream involves the modification of the transport stream using a PID filter. A PID filter 1404 can be implemented as part of the demodulator 1420 of Figure 14 or as part of demultiplexer.

For illustrative purposes, assuming a multi-program transport stream with each program consisting of an I-PIDs for both video and guide, PRED-PID for both video and guide, audio-PID, and data-PID, any packet with a PID that matches any of the PIDs within the desired program as identified by the program mapping table to be received have its PID modified to the lowest video PID in the program (the PID which is referenced first in the program's program mapping table (PMT)). For example, in a program, assuming that a guide slice I-PID is 50, the video slice I-PID is 51 and PRED-PID is 52. Then, the PID-filter modifies the video I-PID and the PRED-PID as 50 and thereby, I- and Predicted-Picture slice access units attain the same PID number and become a portion of a common stream.

As a result, the transport stream output from the PID filter contains a program with a single video stream, whose packets appear in the proper order to be decoded as valid MPEG bitstream.

Note that the incoming bit stream does not necessarily contain any packets with a PID equal to the lowest video PID referenced in the programs PMT. Also note that it is possible to modify the video PID's to other PID numbers than lowest PID without changing the operation of the algorithm.

When the PID's of incoming packets are modified to match the PID's of other packets in the transport stream,

the continuity counters of the merged PID's may become invalid at the merge points, due to each PID having its own continuity counter. For this reason, the discontinuity indicator in the adaptation field is set for any packets that may immediately follow a merge point. Any decoder components that check the continuity counter for continuity is required to correctly process the discontinuity indicator bit.

Figure 16 illustrates the details of this method, in which, it starts at step 1605 and proceeds to step 1610 to wait for (user) selection of two I-PIDs, illustratively two PIDs corresponding to guide and video portion slices, to be received. The I-PIDs, comprising the first picture of a stream's GOP, represents the two streams to be received. Upon detecting a transport packet having one of the selected I-PIDs, the method 1600 proceeds to step 1615.

At step 1615, the PID number of the I-stream is re-mapped to a predetermined number, PID*. At this step, the PID filter modifies all the PID's of the desired I-stream packets to PID*. The method then proceeds to step 1620, wherein the PID number of the predicted picture slice streams, PRED-PID, is re-mapped to PID*. At this step, the PID filter modifies all the PID's of the PRED-PID packets to PID*. The method 1600 then proceeds to step 1625.

At step 1625, the packets of the PID* stream are extracted from the transport stream by the demultiplexer. The method 1600 then proceeds to step 1630, where the payloads of the packets that includes video stream header information and I-picture and predicted picture slices are coupled to the video decoder as video information stream V. Note that the slice packets are ordered in the transport stream in the same order as they are to be decoded, i.e., a guide slice packets of first row followed by video slice packets of first row, second row, and so on. The method 1600 then proceeds to 1635.

At step 1635, a query is made as to whether a different set of (two) I-PIDs are requested. If the query at step 1635 is answered negatively, then the method 1600 proceeds to step 1610 where the transport demultiplexer waits for the next packets having the identified I-PIDs. If the query at step 1635 is answered affirmatively, then the two PIDs of the new desired I-picture is identified at step 1640 and the method 1600 returns to step 1610.

The method 1600 of Figure 16 is used to produce a conformant MPEG video stream by merging the intra-coded slice streams and predictive-coded slice streams before the demultiplexing process.

E3. Recombination Method 3

The third method accomplishes MPEG bitstream recombination by using splicing information in the adaptation field of the transport packet headers by switching between video PIDs based on splice countdown concept.

In this method, the MPEG streams signal the PID to PID switch points using the splice countdown field in the transport packet header's adaptation field. When the PID filter is programmed to receive one of the PIDs in a program's PMT, the reception of a packet containing a splice countdown value of 0 in its header's adaptation field causes immediate reprogramming of the PID filter to receive the other video PID. Note that a special attention to splicing syntax is required in systems where splicing is used also for other purposes.

Figure 17 illustrates the details of this method, in which, it starts at step 1705 and proceeds to step 1710 to wait for (user) selection of two I-PIDs to be received. The I-PIDs, comprising the first picture of a stream's GOP, represents the stream to be received. Upon detecting a

transport packet having one of the selected I-PIDs, the method 1700 proceeds to step 1715.

At step 1715, the I-PID packets are extracted from the transport stream until, and including, the I-PID packet with slice countdown value of zero. The method 1700 then proceeds to step 1720 where the payloads of the packets that includes header information related to video stream and I-picture slice data are coupled to the video decoder as video information stream V. The method 1700 then proceeds to step 1725.

At step 1725, the PID filter is re-programmed to receive the predicted picture packets PRED-PID. The method 1700 then proceeds to 1730. At step 1730, the predicted stream packets, illustratively the PID11 packets of predicted picture slices, are extracted from the transport stream. At step 1735, the payloads of the packets that includes header information related to video stream and predicted-picture data are coupled to the video decoder. At the end of step 1735, a complete GOP, including the I-picture slices and the predicted-picture slices, are available to the video decoder. As the payloads are sent to the decoder in exactly in the order in which the packets arrive at the demultiplexer, the video decoder decodes the recombined stream with no additional recombination process. The method 1700 then proceeds to step 1740.

At step 1740, a query is made as to whether a different I-PID set (two) is requested. If the query at step 1740 is answered negatively, then the method 1700 proceeds to step 1750 where the PID filter is re-programmed to receive the previous desired I-PIDs. If answered affirmatively, then the PIDs of the new desired I-picture is identified at step 1745 and the method proceeds to step 1750, where the PID filter is re-programmed to receive the new desired I-PIDs. The method then proceeds to step 1745,

where the transport demultiplexer waits for the next packets having the PIDs of the desired I-picture.

The method 1700 of Figure 17 is used to produce a conformant MPEG video stream, where the PID to PID switch is performed based on a splice countdown concept. Note that the slice recombination can also be performed by using the second method where the demultiplexer handles the receiving PIDs and extraction of the packets from the transport stream based on the splice countdown concept. In this case, the same process is applied as Figure 17 with the difference that instead of reprogramming the PID filter after "0" splice countdown packet, the demultiplexer is programmed to depacketize the desired PIDs.

E4. Recombination Method 4

For the receiving systems that do not include a PID filter and for those receiving systems in which the demultiplexer can not process two PIDs for splicing the streams, a fourth method presented herein provides the stream recombination. In a receiver that cannot process two PIDs, two or more streams with different PIDs are spliced together via an additional splicing software or hardware and can be implemented as part of the demultiplexer. The process is described below with respect to Figure 18. The algorithm provides the information to the demultiplexer about which PID to be spliced to as the next step. The demultiplexer processes only one PID but a different PID after the splice occurs.

Figure 18 depicts a flow diagram of this fourth process 1800 for recombining the IPG streams. The process 1800 begins at step 1801 and proceeds to step 1802 wherein the process defines an array of elements having a size that is equal to the number of expected PIDs to be spliced. It is possible to distribute splice information in a picture as desired according to slice structure of the picture and

the desired processing form at the receiver. For example, in the slice based streams discussed in this invention, for an I picture, splice information may be inserted into slice row portions of guide and video data. At step 1804, the process initializes the video PID hardware with for each entry in the array. At step 1810, the hardware splice process is enabled and the packets are extracted by the demultiplexer. The packet extraction may also be performed at another step within the demultiplexer. At step 1812, the process checks a hardware register to determine if a splice has been completed. If the splice has occurred, the process, at step 1814, disables the splice hardware and, at step 1816, sets the video PID hardware to the next entry in the array. The process then returns along path 1818 to step 1810. If the splice has not occurred, the process proceeds to step 1820 wherein the process waits for a period of time and then returns along path 1822 to step 1812.

In this manner, the slices are spliced together by the hardware within the receiver. To facilitate recombining the slices, the receiver is sent an array of valid PID values for recombining the slices through a user data in the transport stream or another communications link to the STT from the HEE. The array is updated dynamically to ensure that the correct portions of the IPG are presented to the user correctly. Since the splice points in slice based streams may occur at a frequent level, a software application may not have the capability to control the hardware for splicing operation as discussed above. If this is the case, then, firmware is dedicated to control the demodulator hardware ,for splicing process at a higher rate than a software application can handle.

F. Example: Interactive Program Guide

The video streams representing the IPG may be carried in a single transport stream or multiple transport streams, within the form of a single or multi-programs as discussed below with respect to the description of the encoding system. A user desiring to view the next 1.5 hour time interval (e.g., 9:30 - 11:00) may activate a "scroll right" object (or move the joystick to the right when a program within program grid occupies the final displayed time interval). Such activation results in the controller of the STT noting that a new time interval is desired. The video stream corresponding to the new time interval is then decoded and displayed. If the corresponding video stream is within the same transport stream (i.e., a new PID), then the stream is immediately decoded and presented. If the corresponding video stream is within a different transport stream, then the related transport stream is extracted from the broadcast stream and the related video stream is decoded and presented. If the corresponding transport stream is within a different broadcast stream, then the related broadcast stream is tuned, the corresponding transport stream is extracted, and the desired video stream is decoded and presented.

It is important to note that each extracted video stream is associated with a common audio stream. Thus, the video/audio barker function of the program guide is continuously provided, regardless of the selected video stream. Also note that the teachings of the invention is equally applicable to systems and user interfaces that employs multiple audio streams.

Similarly, a user interaction resulting in a prior time interval or a different set of channels results in the retrieval and presentation of a related video stream. If the related video stream is not part of the broadcast video streams, then a pointcast session is initiated. For this

purpose, the STT sends a request to the head end via the back channel requesting a particular stream. The head end then processes the request, retrieves the related guide and video streams from the information server, incorporates the streams within a transport stream as discussed above (preferably, the transport stream currently being tuned/selected by the STT) and informs the STT which PIDs should be received, and from which transport stream should be demultiplexed. The STT then extracts the related PIDs for the IPG. In the case of the PID being within a different transport stream, the STT first demultiplexes the corresponding transport stream (possibly tuning a different QAM stream within the forward channel).

Upon completion of the viewing of the desired stream, the STT indicates to the head end that it no longer needs the stream, whereupon the head end tears down the pointcast session. The viewer is then returned to the broadcast stream from which the pointcast session was launched.

Although various embodiments which incorporate the teachings of the present invention have been shown and described in detail herein, those skilled in the art can readily devise many other varied embodiments that still incorporate these teachings. An important note is that the method and apparatus described herein is applicable to any number of slice assignments to a video frame and any type of slice structures. The presented algorithms are also applicable to any number of PID assignments to intra-coded and predictive-coded slice based streams. For example, multiple PIDs can be assigned to the predictive-coded slices without loss of generality. Also note that the method and apparatus described herein is fully applicable picture based encoding by assigning each picture only to a one slice, where each picture is encoded then as a full frame instead of multiple slices.